WATER RESOURCES RESEARCH, VOL. 35, NO. 5, PAGES 1605–1618, MAY 1999

Estimation of physical variables from multichannel remotely sensed imagery using a neural network: Application to rainfall estimation

Kuo-lin Hsu, Hoshin V. Gupta, Xiaogang Gao, and Soroosh Sorooshian
Department of Hydrology and Water Resources, University of Arizona, Tucson

Abstract. Satellite-based remotely sensed data have the potential to provide hydrologically relevant information about spatially and temporally varying physical variables. A methodology for estimating such variables from multichannel remotely sensed data is presented; the approach is based on a modified counterpropagation neural network (MCPN) and is both effective and efficient at building complex nonlinear input-output function mappings from large amounts of data. An application to high-resolution estimation of the spatial and temporal variation of surface rainfall using geostationary satellite infrared and visible imagery is presented. Test results also indicate that spatially and temporally sparse ground-based observations can be assimilated via an adaptive implementation of the MCPN method, thereby allowing on-line improvement of the estimates.

1. Introduction and Background

The retrieval of physical variables (such as precipitation, surface soil moisture, etc.) from multichannel remotely sensed information involves several factors that complicate the estimation problem. First, the relationship between the physical variables and the remotely sensed signals may inherently be highly nonlinear and changing in both time and space. Second, in the absence of complete information on the relevant influencing variables, a one-to-one inverse mapping may not be achievable; while auxiliary information from multiple sensors may help to reduce the mapping uncertainty, the trade-off is an increase in the number of independent variables. Third, the sheer volume of remotely sensed data to be processed must be considered. Therefore a suitable retrieval strategy must have the ability to carry out spatially and temporally variable nonlinear transformations of large amounts of information from multiple sensors (satellite, airborne, radar, etc.) in a cost-effective manner.

Artificial neural network (ANN) models have recently found application in several studies involving the processing of remotely sensed imagery, including classification of vegetation types from multichannel image data [*Heermann and Khazenie*, 1992; *Bischoff et al.*, 1992], estimation of snow parameters from passive microwave special sensor microwave imager (SSM/I) imagery [*Tsang et al.*, 1992; *Davis et al.*, 1993], classification of terrain types and land-cover mapping using synthetic aperture radar images [*Yoshida and Omatu*, 1994; *Serpico and Roli*, 1995], identification of cloud segmentation from advanced very high resolution radiometer (AVHRR) data [*Yhann and Simpson*, 1995], retrieval of relative humidity from microwave imagery [*Cabrera-Mercader and Staelin*, 1995], estimation of rainfall from radar observations [*Xiao and Chandrasekar*, 1997], and the retrieval of vapor, liquid, and ice from passive microwave images [*Li et al.*, 1997].

Paper number 1999WR900032.
0043-1397/99/1999WR900032$09.00

The usefulness of ANN models in such applications is due not only to their ability to deduce and model unknown nonlinear structures, but also to their ability to learn from data while adapting to evolutionary system behavior. Important and desirable features of ANN models include (1) the ability to model nonlinear behavior of arbitrary complexity, (2) a parallel computational structure which enables very rapid data processing, and (3) the relative ease of incorporating additional prognostic variables. The ANN model architecture that has been most often applied in remote-sensing applications is the multilayer feedforward neural network (MFNN), also known as the back-propagation network or perceptron. Although the MFNN is capable of providing continuous nonlinear approximations of an input-output mapping to a high level of accuracy, it is notoriously difficult and expensive to train. This is because the cost function in the model parameter space is typically highly nonconvex, characterized by insensitive fault regions, and has large numbers of local minima [*Jacobs*, 1988; *Gori and Tesi*, 1992; *Gupta et al.*, 1997]. In contrast, the less well known counterpropagation network (CPN) architecture typically requires several orders of magnitude fewer training iterations [*Hecht-Nielsen*, 1990], but the accuracy of the function mapping is limited to a piecewise constant approximation (equivalent, in essence, to a lookup table). The CPN is therefore most applicable to discrete classification problems.

This research utilizes a "modified" counterpropagation network (MCPN) architecture developed at The University of Arizona. The MCPN provides accurate approximations of nonlinear spatially and temporally variable input-output function mappings while retaining the desirable rapid training characteristics of the CPN architecture. A preliminary application of the MCPN methodology to rainfall estimation using single-channel satellite infrared imagery was made by *Hsu et al.* [1997]. In this paper, we go further and explain how the methodology can be applied to the general problem of estimating physical variables, describe the mathematical and implementational details of the procedure (section 2), and explain the how and why of preprocessing the large volumes of remotely sensed data (section 3). Finally, a new application study is

presented which demonstrates how the MCPN method is able to synthesize information from multichannel (both infrared and visible) satellite imagery into hourly high-resolution estimates of rainfall (section 4).

2. Artificial Neural Networks

The mathematics of ANN models arose from the study of how biological neural networks enable humans and animals to adaptively learn and perform complex tasks in environments that are fuzzy, noisy, and evolving with time. Although the field can be considered still to be in its infancy, ANN methods have found a variety of practical applications, including time-series forecasting, process control, pattern recognition, hydrological process simulation, and image processing [*Kosko*, 1992; *Vemuri and Rogers*, 1994; *Hsu et al.*, 1995; *Suykens et al.*, 1996]. As mentioned earlier, applications to the physical sciences have increased in number.

Theoretical and practical studies have indicated that the multilayer feedforward neural network (MFNN) is a universal function approximator which is capable of modeling any nonlinear differentiable input-output function to an arbitrary degree of accuracy [*Funahashi*, 1989; *Hornik et al.*, 1990]. The MFNN has therefore been the most widely used ANN architecture. However, the MFNN has several characteristics that make it a less than optimal choice for estimation of physical variables from remotely sensed data. First, optimal training is difficult to achieve; the error function surface of the MFNN to be searched during network training contains numerous local minima and extensive regions of insensitivity, making the training process (search for a set of optimal weights) a relatively slow and computationally expensive endeavor [*Jacobs*, 1988; *Gori and Tesi*, 1992; *Gupta et al.*, 1997]. Second, the MFNN is not a cost-effective way to process large amounts of data. In particular, the case study presented in this paper uses training data consisting of 270,000 patterns of 10 input variables and cannot feasibly be handled using an MFNN architecture. Third, the MFNN is usually considered to belong to the class of "black-box" methods because it does not readily lend itself to interpretation of the logical relationships between the input and output variables.

An alternative ANN architecture, called the counterpropagation network (CPN) [*Hecht-Nielsen*, 1990], is simpler to build and requires less computer time to train than the MFNN, while facilitating conceptual interpretation of the relationships between the input and output model variables. The CPN structure consists of two functional components: The input-hidden portion performs an input classification by mapping the input vectors into a number of clusters, and the hidden-output portion maps the classified inputs into the output variable. This functional separation allows the network training procedure to be simplified. However, because the CPN structure is limited to providing only piecewise constant function approximations, we have developed a modified version that provides accurate and smoothly varying approximations while retaining the training characteristics of the CPN architecture.

2.1. Counterpropagation Network

The counterpropagation network (CPN) introduced by *Hecht-Nielsen* [1987, 1988] is a hybrid three-layer network. The input-hidden portion of the network performs an unsupervised clustering using a technique called the self-organizing feature map (SOFM) [*Kohonen*, 1982]; the SOFM provides the nonlinear portion of the input-output mapping. The hidden-output portion consists of a supervised linear mapping called a Grossberg Linear Network (GLN) [*Grossberg*, 1969]. The function of the SOFM is, as the name suggests, to achieve a classification of the input variables on the basis of a set of representative training (input) data vectors. This organizes the input variables onto a discrete one- or two-dimensional map which preserves the topological order of the inputs [*Hecht-Nielsen*, 1990]. The function of the GLN, on the other hand, is to establish a linear predictive association between the input clusters and the output variable by means of a supervised training procedure. Because the mapping problem is partitioned into two sequential operations (an unsupervised SOFM and a supervised GLN, with the GLN being conditioned on the results of the SOFM), the CPN is relatively quick and inexpensive to build and train, making it particularly suitable for situations that require processing of large volumes of data.

The CPN model has two sets of weights, w_{ji} (which define the cluster centers of the SOFM) and v_{kj} (which correspond to the GLN), which must be trained. For each input vector x_i the "distance" between the (normalized) input vector and each hidden layer (SOFM) node is computed as

$$d_j = \left[\sum_{i=1}^{n0} (x_i - w_{ji})^2 \right]^{0.5} \qquad j = 1, \cdots, n_1 \tag{1}$$

The input vector is then assigned to the hidden node, $j = I_c$, which is closest ($d_c = \min(d_j)$, $j = 1, \cdots, n_1$), and the output from the node is computed using a trivial form of the GLN as

$$\begin{aligned} y_j &= 1.0 \qquad \text{if } j = I_c \\ y_j &= 0 \qquad \text{otherwise} \end{aligned} \tag{2}$$

The estimate of the GLN output variable z_k is then computed as

$$z_k = v_{kj} \qquad j = I_c \tag{3}$$

From (1)–(3) it can be seen that input vectors with slight differences in "distance" but which classify into the same cluster will result in identical estimated outputs; that is, the region of the input space corresponding to a particular SOFM node is assigned a constant output value equivalent to the mean of all training outputs associated with that input space. The output mapping is therefore neither continuous nor smooth, and the accuracy of the function approximation is directly proportional to the number of nodes, n_1, in the SOFM layer. The next section discusses how modifying the GLN to a suitable piecewise-linear interpolation function enables us to increase the accuracy of the CPN function approximation without increasing the number of SOFM nodes.

2.2. Modified Counterpropagation Network

The architecture of the modified counterpropagation network (MCPN) system is illustrated in Figure 1. Notice that (1) the hidden nodes and output nodes are arranged as a pair of node-to-node coincident two-dimensional matrices and (2) each output node is connected to a set of hidden nodes in the neighborhood of the coincident hidden node. The SOFM output y_j is computed as follows (compare with (2)) to represent the degree of closeness of an input vector to an SOFM node:

$$\begin{aligned} y_j &= 1 - d_j \qquad \text{for } j \in \Omega \\ y_j &= 0 \qquad \text{otherwise} \end{aligned} \tag{4}$$

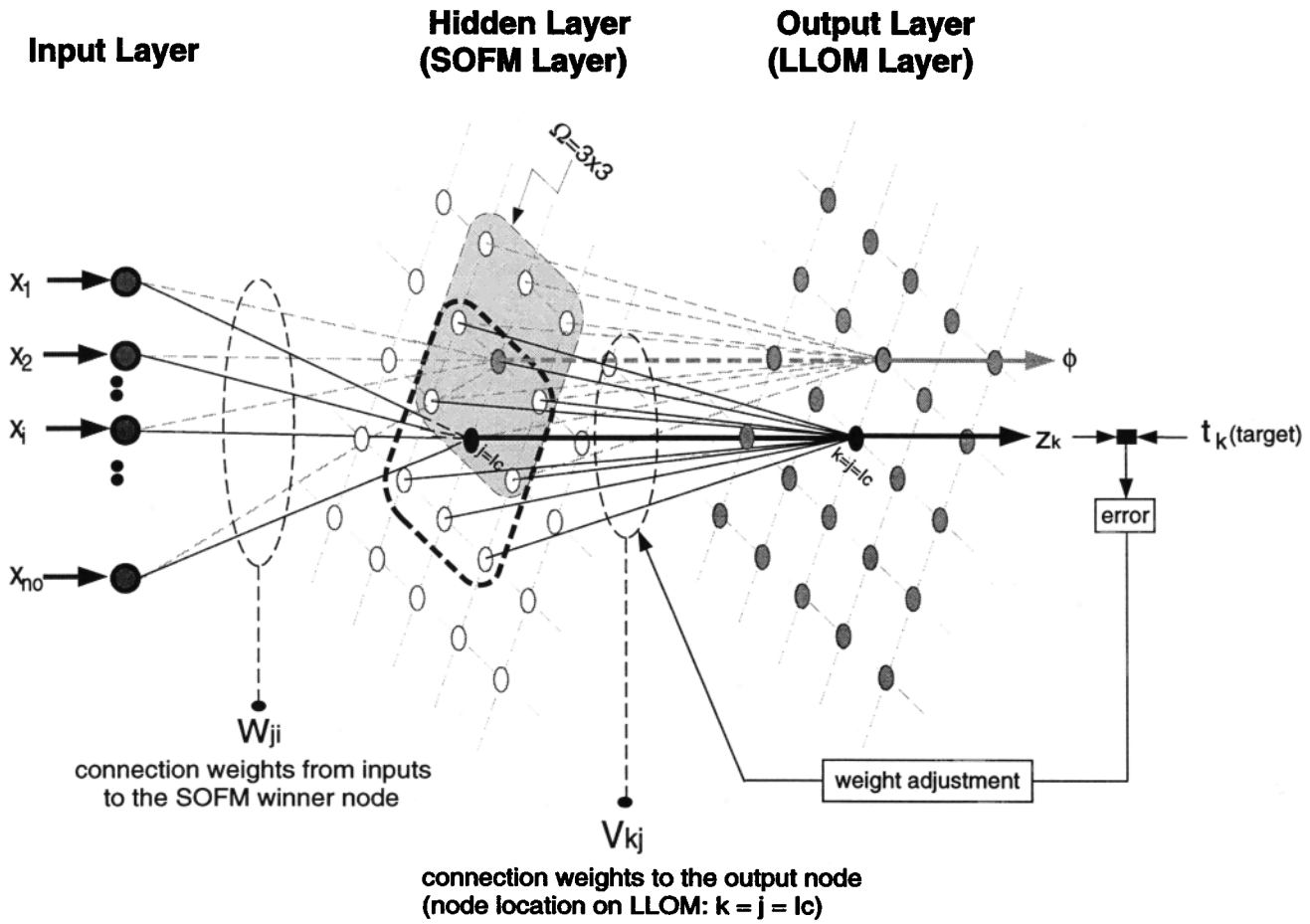


Figure 1. The architecture of the modified counterpropagation network.

where Ω is centered on I_c and is a selected neighborhood surrounding the winner node. To ensure uniqueness (nondegeneracy) of the hidden-output mapping, the size (number of nodes) of the neighborhood Ω should be equal to or greater than the dimension of the input variable space. The estimate of the output is computed using a Local Linear Output Mapping (LLOM) (a form of the Grossberg Linear Network (GLN)) as the weighted summation of the outputs (compare with (3)) belonging to the Ω neighborhood of the selected SOFM node:

$$\begin{aligned} z_k &= \Sigma\ v_{kj} y_j \qquad \text{if } j \in \Omega \wedge k = I_c \\ z_k &= \varnothing \qquad\qquad \text{otherwise} \end{aligned} \tag{5}$$

With this structure, different input vectors that cluster into the same SOFM node will generate different sets of hidden outputs y_j and will therefore result in different output estimates. Further, because the output nodes form a two-dimensional matrix that is node-to-node coincident with the hidden-node matrix, each output node has a unique set of connections to the SOFM layer, so that the calculation of connection weights for each output node is independent and efficient, and the mapping accuracy is increased.

The procedure for training the MCPN weights w_{ji} and v_{kj} is quite straightforward and is discussed below. Note that while the input-hidden (SOFM) weights w_{ji} must be trained off-line, the hidden-input (LLOM) weights v_{kj} can be first initialized through off-line training and then iteratively refined by a feedback process (on the basis of observed performance) during on-line operation of the network. This allows the model to adaptively track input-output mappings that may vary with geographical location and with season. The case study presented in section 4 illustrates the performance of the MCPN model under both off-line and on-line training of the hidden-output (LLOM) weights.

2.3. Network Training

Training of the MCPN model is conducted in two stages. The SOFM (w_{ji}) weights are trained only in "off-line" mode (the weights are not changed during on-line operation), while the LLOM (v_{kj}) weights can be conducted during both the off-line and on-line modes of operation. The procedure for training the SOFM weight vectors $\mathbf{w}_j = [w_{j1}, \cdots, w_{jn0}]$ $j = 1, 2, \cdots, n_1$ is as follows:

Step 1: Randomly initialize the connection weights for each hidden node, $\mathbf{w}_j(0), j = 1, 2, \cdots, n_1$ ($w_{ji} \in [0, 1], i = 1, 2, \cdots, n_0$).

Step 2: Compute the "distance" d_j between a (normalized) training input vector $\mathbf{x}_i$ and each hidden-layer node according to (1).

Step 3: Select the winning node I_c, so that $d_c = \min(d_j)$, $j = 1, \cdots, n_1$.

Step 4: For all hidden nodes in the neighborhood of the winning node, refine their connection weights as follows (the nodes outside $\Lambda_c(t)$ remain the same as before):

$$\mathbf{w}_j(t) = \mathbf{w}_j(t-1) + \eta(t)[\mathbf{x}_j(p) - \mathbf{w}_j(t-1)] \quad \text{if } j \in \Lambda_c(t)$$
$$\mathbf{w}_j(t) = \mathbf{w}_j(t) \quad \text{otherwise} \tag{6}$$

where t is the training iteration index, $0 < \eta(t) < 1$ is the learning rate, and $\Lambda_c(t)$ defines the neighborhood size around the winning node I_c.

Step 5: Decrease the learning rate $\eta(t)$ and the neighborhood size $\Lambda_c(t)$:

$$\eta(t) = \eta_0[1.0 - (t/T)] \tag{7}$$

$$\Lambda(t)_c = \Lambda_0[1.0 - (t/T)] \tag{8}$$

where T is the total number of training iterations and the initial settings are $\eta_0 = 0.2 \sim 0.5$ and $\Lambda_0 = n_1/2$.

After completing steps 1–5 through the given training (input) data set, repeat steps 2–5 several times until the values of the weights w_{ji} stabilize.

The procedure for training the LLOM weights v_{kj} after completion of the SOFM training is as follows. Let $\mathbf{t}(\mathbf{p}) = [t_1[p], t_2[p], \cdots, t_{n2}[p]]$ be the output "target" data corresponding to the input data $\{\mathbf{x}(p)\}$ and define the model error as a function of the LLOM weights v_{kj}. We determine the optimal values of v_{kj} by minimizing the error function F_k associated with each SOFM node, where F_k is defined over the total number of patterns m_k as

$$F_k = \frac{1}{2}\sum_{p=1}^{m_k}[t_k(p) - z_k(p)]^2 \tag{9}$$

Notice that the error functions F_k for each output node are independent of the others and can be optimized separately. During off-line training the weights v_{kj} for each output node can be calculated by batch processing of the entire training data set. The linear relationship between z_k and y_j makes the calculation of the weights v_{kj} a simple matter of solving the following equation:

$$\frac{\partial F_k}{\partial v_{kj}} = -\sum_{p=1}^{m}[t_k(p) - z_k(p)]y_j(p) = 0 \tag{10}$$

which can be rearranged to give

$$\mathbf{v_k} = [\mathbf{R}]^{-1}[\mathbf{Q}] \tag{11}$$

where

$$R_{jh} = \sum_{p=1}^{m_k} y_j(p)y_h(p) \tag{12}$$

$$Q_j = \sum_{p=1}^{m_k} t_k(p)y_j(p) \tag{13}$$

During on-line training, only a limited number of output observations may be available, and a simple "sequential" training procedure therefore is used; that is, the weights corresponding to each output node are incrementally adjusted in the directions of the negative gradients of their error functions:

$$\Delta v_{kj} = -\frac{\partial F[v]}{\partial v_{kj}} = [t_k(p) - v_{kj}y_j(p)]y_j(p) \tag{14}$$

Using the Widrow-Hoff learning rule,

$$v_{kj}(t) = v_{kj}(t-1) + \beta[t_k(p) - z_k(p)]y_j(p) \quad \text{if } j \in \Omega \wedge k = I_c$$
$$v_{kj}(t) = v_{kj}(t-1) \quad \text{otherwise} \tag{15}$$

The learning step size β is chosen to be between 0 and 1 (e.g., $\beta = 0.1$). As shown in (15), only the weights associated with the neighborhood Ω of the "winner" node are varied for a data pattern; the other weights remain the same.

3. Preprocessing of Training Data

A characteristic of remotely sensed image data is the extremely huge volume of bits of information that must be stored and manipulated. The key to efficiently building input-output mappings using such data is the recognition that the data contain a great deal of redundant information. Further, the information content of the data is typically skewed to give a nonuniform representation of the input-output behaviors of interest. As an example, consider the case of estimating rainfall from satellite data. Because the data are collected on a 30-min time interval, most of the satellite images (both infrared and visible) correspond to nonraining situations, while images corresponding to occurrence of rainfall occur only for short time periods and small regions of the entire temporal and spatial regime covered by satellites. Experiments have shown that if the data are passed through a preprocessing filter before use, the identified MCPN model provides more accurate (less biased) estimates of the dependent variable. Further, the computation time is dramatically reduced.

The preprocessing procedure follows three steps (see Figure 2): (1) A regular grid system with an appropriate spatial resolution is overlayed on the input-variable space, thereby uniformly dividing the input space into equal-sized grid boxes; (2) each input data vector is classified as belonging to one of the grid boxes; (3) a representative input training data set is constructed by selecting a fixed number of data points (one or more) from each grid box. Note that in the case of constructing a more detailed representation of some portion of the input space, we can select a higher density of data sets from that region. A simple example that clearly demonstrates the data preprocessing and construction of a MCPN model for a two-input mathematical function is presented in the appendix.

4. Case Study: Rainfall Estimation Using Geostationary Satellite Multichannel Imagery

Rainfall is the major input to the land phase of the hydrological cycle. Good quality rainfall estimates are essential to both short-term hydrological forecasting and long-term water resources systems planning. While ground-based rainfall measurements provided by rain gauges and radar are readily available for some regions of the world, reliable measurements are difficult to obtain for remote parts of the world and for regions such as mountains and oceans. Satellite-based techniques can provide improved rainfall estimates for such regions, thereby helping to fill the gaps in global rainfall coverage.

Methods for global scale rainfall estimation using infrared (IR) data collected by geostationary satellites have been under investigation since the 1960s. Such methods can be classified into grid/pixel-based and cloud-based approaches [*Negri and*

Adler, 1987a, b]. Studies have shown that the grid/pixel methods such as the global precipitation index (GPI) methods provide fairly accurate estimates of rainfall when accumulated to a monthly time step and 5° × 5° (~500 × 500 km) area but are unable to provide accurate estimates at finer spatial and temporal resolutions [*Arkin*, 1979; *Negri and Adler*, 1987a, b]. Meanwhile, there have been several developments of the Griffith-Woodley cloud-patch technique (GWT) [*Griffith et al.*, 1978], including the Negri-Adler-Wetzel technique (NAWT) approach [*Negri et al.*, 1984] and the convective-stratiform technique (CST) [*Adler and Negri*, 1988]. The latest version of the CST algorithm is able to provide good estimates of hourly rainfall after averaging the results to a 128 × 128 km resolution grid over the Florida peninsula. Further improvements of the CST algorithm rely on the use of 86-GHz microwave channel data from the polar-orbiting special sensor microwave imager (SSM/I) [*Negri and Adler*, 1993].

Various studies have also attempted to use multichannel information from visible (VIS) and IR imagery in rainfall estimation [*King et al.*, 1995; *Negri and Adler*, 1987a, b; *Grassotti and Grand*, 1994]. While both IR and VIS imagery are indirectly related to surface rainfall, they provide complementary information that is useful in discriminating nonraining and raining portions of the cloud-covered areas. However, while IR imagery is available 24 hours a day, VIS imagery is available only during the daytime and must be adjusted (normalized) to take account of the position of the Sun [*Tsonis and Isaac*, 1985; *King et al.*, 1995]. *Grassotti and Grand* [1994] used a clustering scheme to analyze VIR, IR, and rainfall rates provided by a numerical weather prediction model for the 4-day Algorithm Intercomparison Program (AIP-1) data testing period and showed that the combined information is capable of providing accurate estimates of hourly rainfall rate. *King et al.* [1995] reported improved performance over the Japanese Islands using combined VIS and IR information. However, *Negri and Adler* [1987a, b] obtained only marginal improvements when incorporating VIS and IR imagery into both the grid-based and cloud definition approaches during the second Florida Area Cumulus Experiments (FACE-2).

In contrast with the rainfall estimation algorithms mentioned above, the MCPN model provides a generalized framework that is suitable for estimating nonlinear functional forms from multivariate data. The following case study illustrates the usefulness of the MCPN model by using it to estimate daytime rainfall (hourly and accumulated monthly) at relatively high spatial resolution from two channels of satellite imagery data: infrared (IR) and visible (VIS).

This case study was conducted using data collected over the Japanese Islands (see Figure 3) by the First Algorithm Intercomparison Project (AIP-1) of the Global Precipitation Climatology Project (GPCP) supported by the World Climate Research Programme (WCRP). Hourly ground-based rainfall data (from rain gauges and radars) and satellite image data are available for the June (JUN/89) and July 15 to August 15 (JUL-AUG/89) 2-month periods during the summer of 1989. Note that the JUN/89 period was dominated by frontal rainfall, while the JUL-AUG/89 period was dominated by convective rainfall. The ground-based data were collected by a network of 1300 rain gauges and 15 precipitation radars operated by the Japanese Automated Meteorological Data Acquisition System (AMeDAS) supported by the Japanese Meteorological Agency (JMA). The satellite-based remotely sensed IR brightness temperature and VIS albedo images were collected by the Geostationary Meteorological Satellite (GMS) [*Arkin and Xie*, 1994]. The VIS imagery used in this study was normalized using a Sun angle adjustment [*King et al.*, 1995]. In this case study, only the daytime data (0000–0700 UTC) from JUN/89 were used to train the MCPN model. The identified model was then tested and evaluated on both of the month-long daytime testing periods (JUN/89 and JUL-AUG/89).

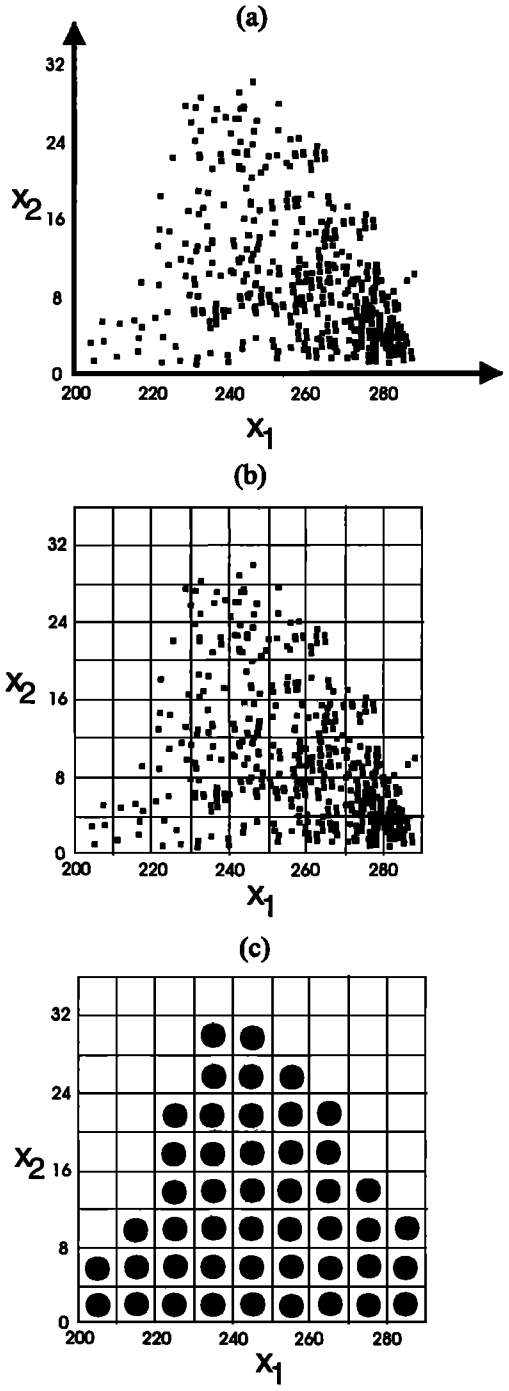


Figure 2. An example of data filtering: (a) distribution of the original data in the (x_1, x_2) input space, (b) setting up a data filter using a regular grid, and (c) selection of the grid center for each grid having at least one data point.

The architecture of the MCPN model used for this study is displayed in Figure 1. The SOFM and LLOM layers each

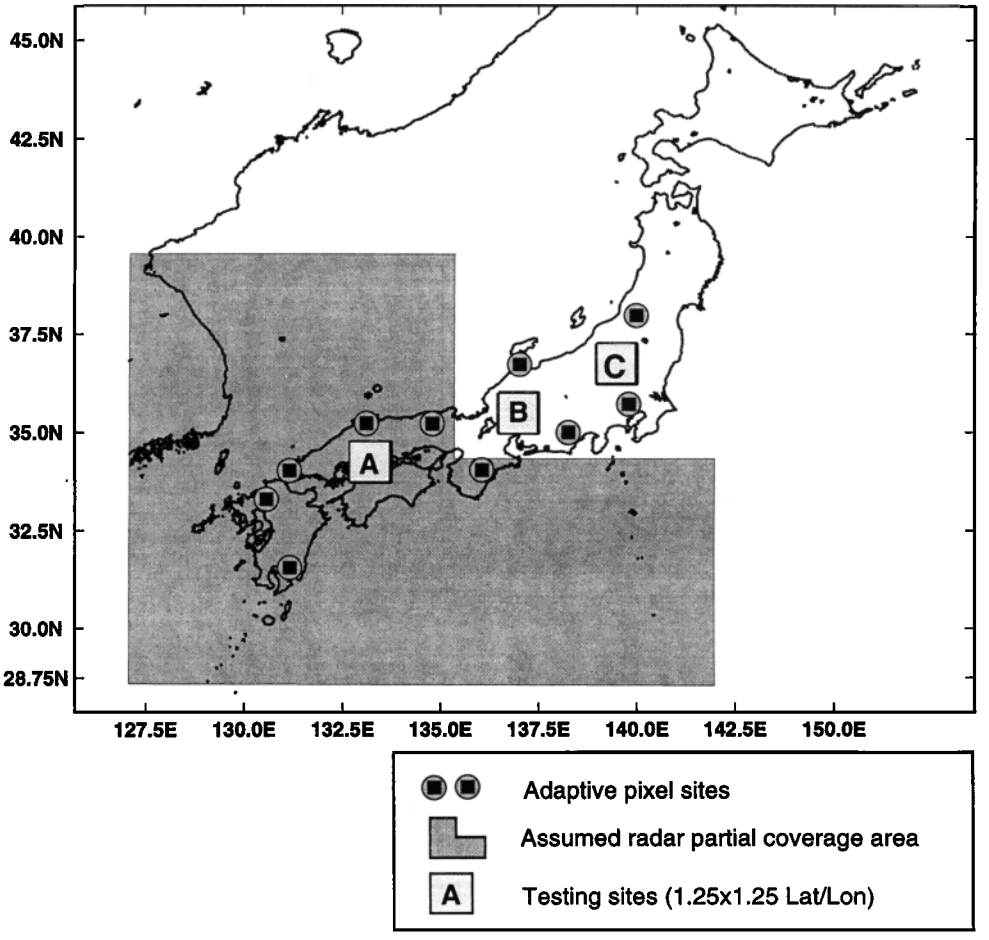


Figure 3. Japanese Islands study area. Hourly time series are generated for 1.25° × 1.25° latitude-longitude regions A, B, and C. The "adaptive pixel site" markers and shaded area indicate the regions from which data are used for adaptive parameter updating.

consist of a matrix of 15 × 15 (= 225) nodes. Each LLOM (output layer) node is connected to a $\Omega = 3 \times 3$ node neighborhood of the coincident SOFM (hidden-layer) node. The input (IR and VIS imagery) and output (rainfall rate) data were prepared to an image pixel resolution of 0.25° × 0.25° latitude-longitude and a hourly timescale. Ten attributes were selected to represent the informative characteristics of the input data (see Table 1). The unfiltered data consisted of more than 270,000 patterns during JUN/89; after preprocessing, only 10,000 patterns were selected (~3.7%) and used for training the SOFM portion of the MCPN model. Next, the LLOM portion of the model was trained in off-line mode by using the full set of 270,000 patterns from the JUN/89 data (frontal rainfall regime). This will be called the "fixed-parameter" case, and results will be presented for both the "calibration period" (JUN/89) and the "evaluation period" (JUL-AUG/89). In the second phase the fixed-parameter model was used as "initial conditions" (a priori estimates), and the LLOM portion of the

Table 1. Infrared and Visible Imagery Input Feature Variables Used for Construction of the Modified Counterpropagation Network Model for Precipitation Estimation

Input Variable	Features
1	IR brightness temperature of the calculation pixel
2	mean IR temperature of the 3 × 3 pixels centered at calculation pixel
3	standard deviation of IR temperature of the 3 × 3 pixels centered at calculation pixel
4	mean IR temperature of the 5 × 5 pixels centered at calculation pixel
5	standard deviation of the 5 × 5 IR pixels centered at calculation pixel
6	VIS brightness temperature of the calculation pixel
7	mean VIS temperature of the 3 × 3 pixels centered at calculation pixel
8	standard deviation of VIS temperature of the 3 × 3 pixels centered at calculation pixel
9	mean VIS temperature of the 5 × 5 pixels centered at calculation pixel
10	standard deviation of the 5 × 5 VIS pixels centered at calculation pixel

IR, infrared; VIS, visible.

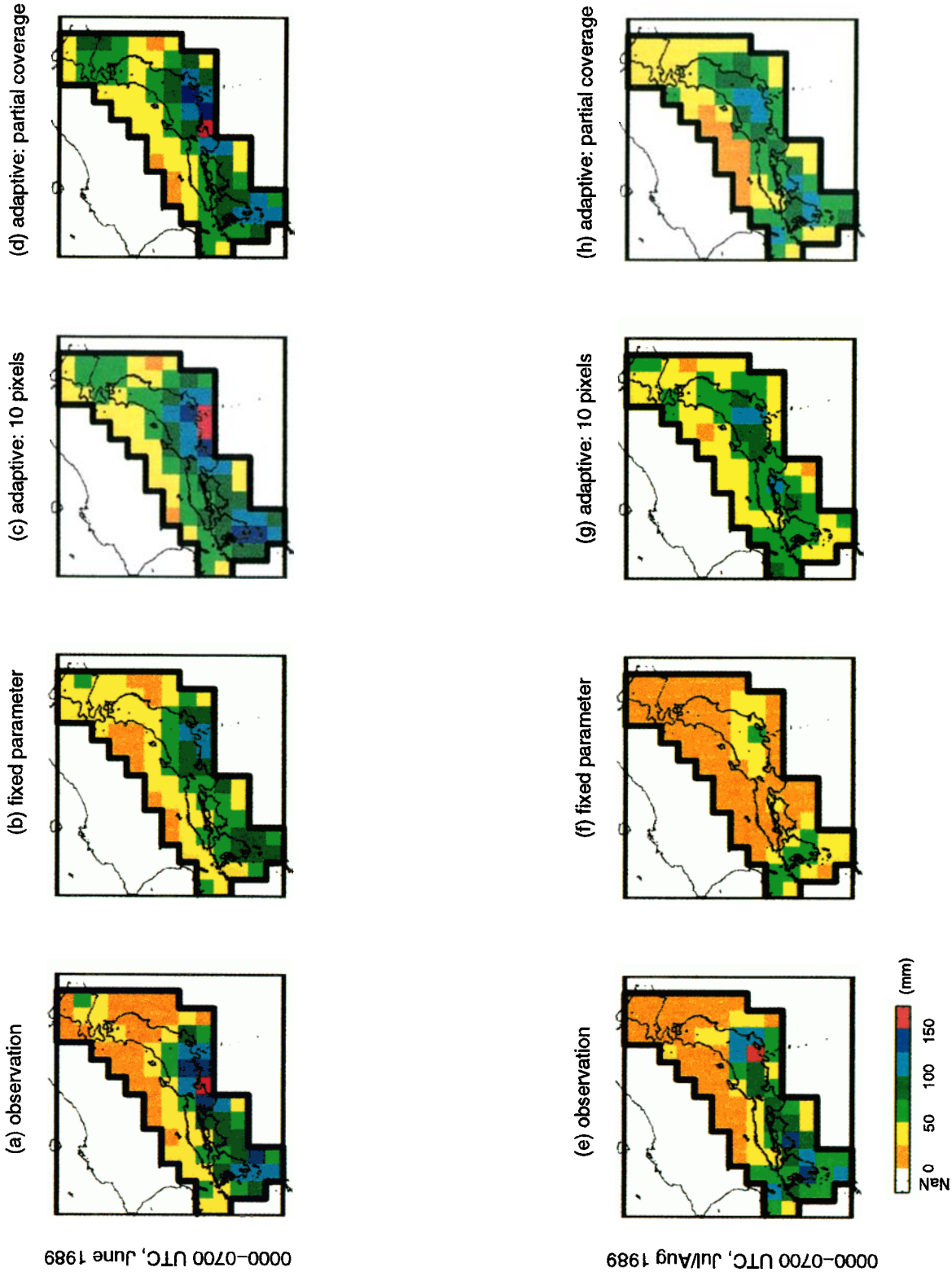


Plate 1. Observed and estimated monthly accumulated daytime rainfall distributions at 1.25° × 1.25° latitude-longitude resolution. The top row is for June 1989 (JUN/89), and the bottom row is for July 15 to August 15, 1989 (JUL-AUG/89). Plots from left to right are ground-based observations, estimates from the fixed-parameter model, estimates after parameter updating using limited data, and estimates after parameter updating using partial-coverage data.

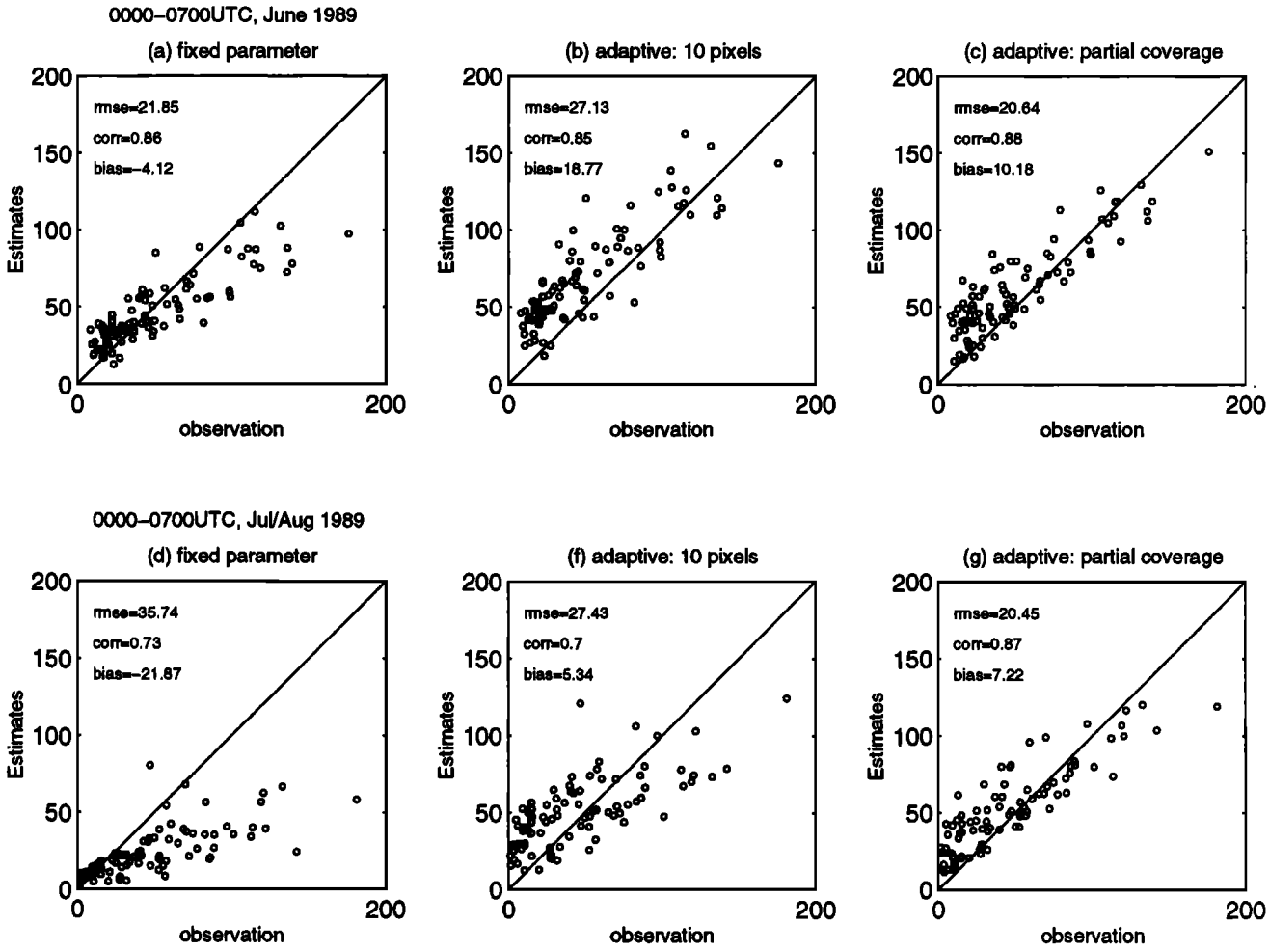


Figure 4. Observed and estimated monthly accumulated daytime rainfall scatterplots. The top row is for JUN/89, and the bottom row is for JUL-AUG/89. Plots from left to right are results using the fixed-parameter model, results after parameter updating using limited data, and results after parameter updating using partial-coverage data.

model (hidden-output weights v_{kj}) was trained in on-line mode (i.e., iterative parameter updating) sequentially through first the JUN/89 (frontal rainfall regime) and then the JUL-AUG/89 periods (convective rainfall regime). Two on-line cases were considered: limited data and partial-coverage data. In the first (limited data) case, only the observation data from 10 randomly selected land-based pixels (marked using square-within-circle markers in Figure 3) were used for parameter updating; this simulates the availability of rain-gauge information. In the second (partial-coverage data) case, only the observation data corresponding to the shaded area in Figure 3 were used for parameter updating; this simulates the availability of only "partial-coverage" by the radar (or radar–rain-gauge composite) information. These will be called the "adaptive-parameter" cases and, again, results will be presented for both data periods.

The training parameters for this case study were set as follows: (1) initial size of training neighbors, $\Lambda_0 = 6$, (2) the initial learning rate, $\eta_0 = 0.5$, and (3) the maximum number of training iterations, $T = 6000$. The learning rate $\eta(t)$ was reduced in a linear manner (equation (6)) during training until reaching the value $\eta(t) = 0.02$, when it was fixed at 0.02. The LLOM layer parameters were trained using linear least squares and a 3 × 3 neighborhood size around each winner node. Hence, although the entire set of 270,000 patterns was used to train the Grossberg layer parameters, the size of the matrix **R** (Equation (11)) to be inverted was only 9 × 9 in size, resulting in a relatively low computational cost. In addition, no computations were performed for SOFM nodes triggered by less than 10 patterns.

4.1. Estimating Accumulated Monthly Daytime Rainfall

Estimates of aggregate monthly rainfall are important for water budget studies. To examine the performance of the model in estimation of aggregate monthly daytime rainfall, the hourly 0.25° × 0.25° latitude-longitude model estimates and observed data were accumulated for entire JUN/89 and JUL-AUG/89 periods and aggregated up to a spatial resolution of 1.25° × 1.25° latitude-longitude (~125 × 125 km). The results are presented in Plate 1; the top row of plots corresponds to JUN/89, and the bottom row of plots corresponds to JUL-AUG/89. In each row the first (leftmost) plot is constructed from the observed rainfall data, the second plot presents the fixed-parameter model-estimated values, the third plot presents the adaptive-parameter model-estimated values based on "limited data," and the fourth plot presents the adaptive-

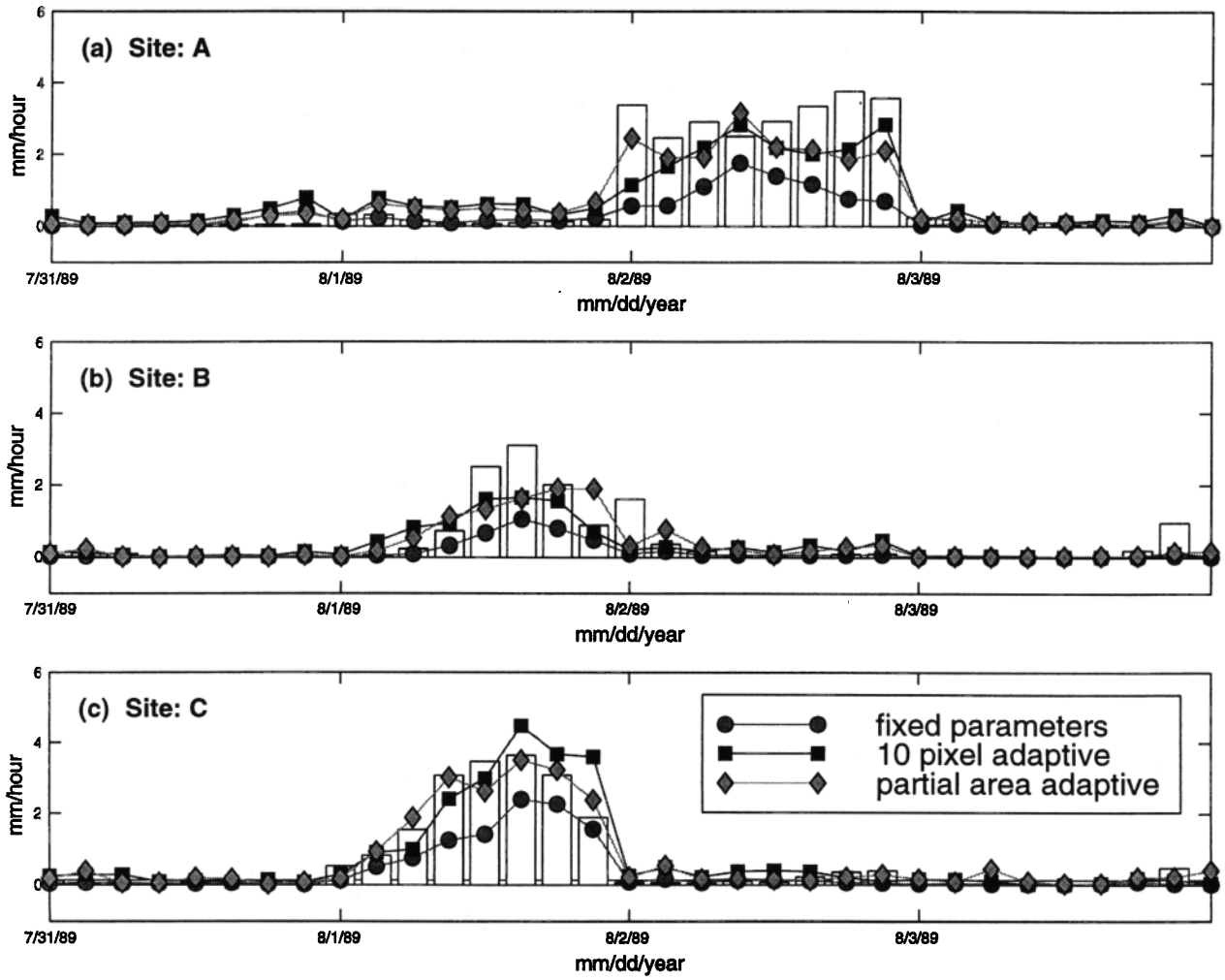


Figure 5. Observed and estimated hourly daytime rainfall time series for sites A, B, and C (see Figure 3); bars indicate observations, circles denote estimates using the fixed-parameter model, squares denote estimates using the limited-data adaptive-parameter model, and diamonds indicate estimates using the partial-coverage data adaptive parameter model.

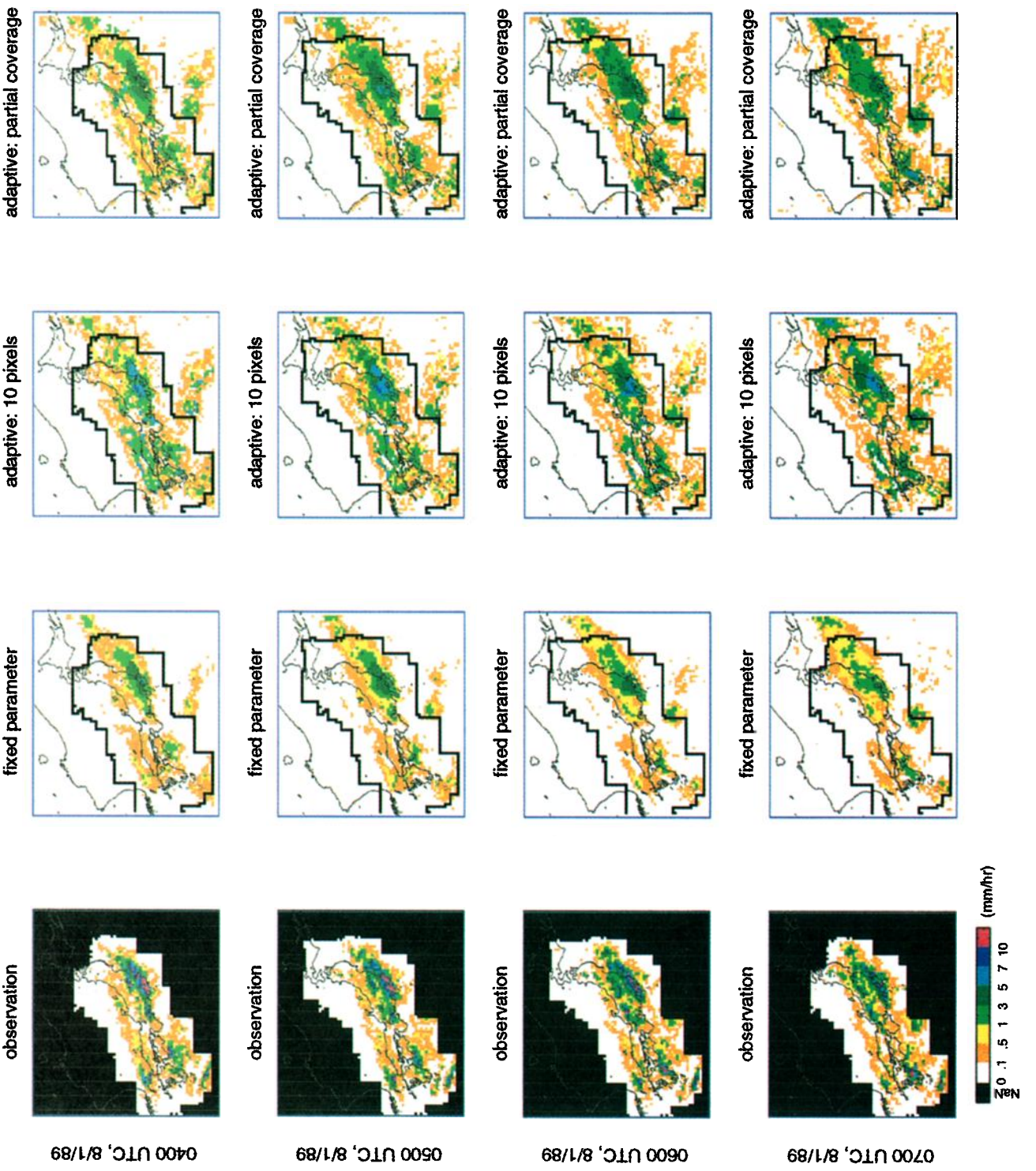
parameter model-estimated values based on "partial-coverage data." Notice that the MCPN model does a generally good job in mapping the spatial distribution of accumulated rainfall. However, the adaptive-parameter results tend to be superior to the fixed-parameter results in terms of matching actual rainfall magnitudes. Figure 4 presents the results in terms of scatter-plots; again, the top and bottom rows represent JUN/89 and JUL-AUG/89, respectively.

The results depicted by Plate 1 and Figure 4 may be summarized as follows. For the JUN/89 calibration period, (1) the fixed parameter model tends to overestimate in the low-rainfall region and underestimate in the high-rainfall region (Plate 1b

Table 2. Evaluation Statistics for Modified Counterpropagation Network Model Adjusted Using Limited Available Ground-Based Data

Testing Sites	Type of Simulation	RMSE	CORR	BIAS
Site A	fixed parameter	1.13	0.81	−0.51
	10 point adaptive	0.67	0.91	−0.08
	partial area adaptive	0.61	0.94	−0.12
Site B	fixed parameter	0.63	0.91	−0.28
	10 point adaptive	0.46	0.86	−0.07
	partial area adaptive	0.49	0.79	−0.04
Site C	fixed parameter	0.59	0.94	−0.26
	10 point adaptive	0.43	0.94	0.11
	partial area adaptive	0.25	0.98	0.05

RMSE, root mean square error; CORR, correlation coefficient; BIAS, bias estimates.


observation
fixed parameter
adaptive: 10 pixels
adaptive: partial coverage
0400 UTC, 8/1/89
0500 UTC, 8/1/89
0600 UTC, 8/1/89
0700 UTC, 8/1/89
NaN
0 .1 .5 1 3 5 7 10
(mm/hr)

and Figure 4a), (2) adaptive training using limited data shows a deterioration in performance over the fixed-parameter model (Plate 1c and Figure 4b), and (3) adaptive training using partial-coverage data shows improvement in performance over the fixed-parameter model (Plate 1d and Figure 4c). For the JUL-AUG/89 evaluation period, (1) the fixed-parameter model significantly underestimates the rainfall (Plate 1f and Figure 4d), (2) adaptive training using limited data shows improvement in performance over the fixed-parameter model (Plate 1g and Figure 4e), and (3) adaptive training using partial-coverage data shows even more improvement (Plate 1h and Figure 4f).

These results are easy to understand when we consider the fact that the JUN/89 period rainfall is generated by a frontal mechanism, while the JUL-AUG/89 period rainfall is convective [*Arkin and Xie*, 1994]. Because the JUN/89 data were used for off-line "calibration" of the model, the initial parameters represented a "good" model for the JUN/89 period, and adaptive training using only a limited amount of data actually caused the model parameters to become biased. However, when the more representative partial-coverage set of data was used for adaptive training, the model performance improved. In each case, the estimated parameters are appropriate for estimating frontal rainfall. When this "frontal-rainfall model" is used during JUL-AUG/89, which is dominated by convective rainfall, the parameters are incorrect, and rainfall is underestimated. However, when the parameters are updated using on-line training, the performance of the model is significantly improved, even when only limited data are used for updating.

4.2. Estimating Time Series of Hourly Rainfall

Time series estimates of rainfall at high temporal resolution are important for a host of hydrological applications ranging from flood forecasting to water resources system planning. To examine the performance of the model in estimation of hourly daytime rainfall, we consider the three 1.25° × 1.25° latitude-longitude blocks denoted as A, B, and C in Figure 3, representing three different regions over the Japanese Islands; in particular, region B is over the central mountainous region. Results are presented for 4 consecutive days (the 8 daylight hours for each of days July 31, 1989, through August 3, 1989) of the JUL-AUG/89 period during which intense rainfall was observed on the ground (see Figure 5 and Table 2). In summary, the fixed parameter model simulates the overall pattern of the rainfall time series reasonably well (correlations above 0.8) but tends to underestimate rainfall magnitudes; this is consistent with the discussion presented earlier: the fixed-parameter model is tuned to frontal rainfall, while the July 31 to August 31, 1989, period is dominated by convective events. However, adaptive estimation of the model parameters quickly removes this bias, even when the limited data scenario is used. Overall, the daytime hourly rainfall estimates provided by the MCPN model appear to be quite good.

Plate 2. (opposite) Observed and estimated hourly rainfall distributions at 0.25° × 0.25° latitude-longitude resolution. Each row is for a successive hour on August 1, 1998. Plots from left to right are ground-based observations, estimates from the fixed-parameter model, estimates after parameter updating using limited data, and estimates after parameter updating using partial-coverage data.

4.3. Estimating Spatial Distributions of Hourly Rainfall

High-resolution maps of spatially distributed hourly rainfall are important for a number of different applications, including coupled land-atmosphere modeling, distributed hydrological modeling, agricultural and land resource management, fire management, urban hydrology, etc. To examine the performance of the model in providing accurate estimates of the spatial distribution of hourly daytime rainfall, we consider four consecutive hourly periods (0400–0700 UTC) for August 1, 1989, during which significant rainfall was observed (see Plate 2). Note that the results are presented at the relatively high resolution (0.25 × 0.25 Lat/Lon) of the model computation which corresponds approximately to pixels of size 25 km × 25 km. Each row in Plate 2 represents a different hour, and the successive columns represent the observed data, fixed-parameter case, adaptive limited-data case, and adaptive partial-coverage data case, respectively. Examining the observed data, we see that the higher-intensity rainfall tends to be concentrated in the central and southern parts of the Japanese Islands; these areas apparently are influenced by the orographic effects of the mountains. Although the fixed-parameter model tends to seriously underestimate the rainfall intensity, it is clear that the spatial pattern of rainfall is well estimated. Again, the use of limited-data adaptive-parameter updating results in improvements in the estimation of rainfall intensity, particularly in the regions of peak rainfall. The partial-coverage data adaptive parameter updating also has resulted in improvements in the estimation of rainfall intensity, but the peak rainfall over the mountains is underestimated; apparently, the region of partial coverage has not provided sufficient information for estimating the higher-rainfall rates in this area, and supplementary information (perhaps using rain gauges) might be beneficial. However, the level of model performance is very encouraging.

5. Summary and Future Directions

This paper has presented an efficient Artificial Neural Network (ANN) approach for the estimation of hydrologically relevant physical variables from multichannel remotely sensed data. The methodology is applicable whether or not the physical variables are constant or evolving with time. The modified counterpropagation network (MCPN) model presented here consists of two components: a self-organizing feature map (SOFM) that constructs a discrete nonlinear classification of the input data space and a local linear output map (LLOM) that constructs a piecewise linear input-output model. The two-component structure of this ANN model permits accurate input-output transformations to be identified in an efficient manner; that is, training of the network is simple to achieve and requires only small amounts of computational resources. We have also proposed a method for preprocessing the data to reduce the computer time required to train the model. For example, in the precipitation estimation example presented here, the initial identification of the SOFM required about 24 hours on a Sun Sparkstation 10, and the processing of each set of hourly images for adaptive estimation of the model parameters required only about 20 s.

The potential usefulness of the method was illustrated by applying it to the difficult problem of estimating rainfall by using satellite-based infrared and visible imagery. Consistent with the findings of other researchers, our results suggest that the functional relationship relating satellite IR and VIS imag-

ery to rainfall rate may be temporally (and probably spatially) variable; that is, a fixed set of parameters calibrated from one rainfall period (JUN/89) which was dominated by frontal rainfall activity was found to be incapable of providing adequate estimates for another rainfall period (JUL-AUG/89) which was dominated by convective rainfall activity. However, we showed that if limited amounts of surface measurements of rainfall or partial-radar-coverage rainfall data (such as in remote or inaccessible regions) are available, the parameters of the model can be adaptively adjusted on-line to provide improved rainfall estimates. Further, the method was able to provide reasonable estimates of the spatial patterns of accumulated monthly rainfall, hourly rainfall time series over a small region, and spatial distributions of hourly rainfall. Given further development, the method shows promise for providing satellite-based estimates of rainfall for a wide variety of hydrologic applications.

A few additional issues related to the applicability of the MCPN method bear additional discussion here. First, let us consider the factors that potentially influence model accuracy: nonstationary input-output mappings, insufficient information, and insufficient data. Clearly, if the physical input-output transformation is temporally or spatially varying, it will be difficult to estimate an accurate and universally applicable model by using data collected at only one site and during a limited time period. The adaptive feature of the MCPN algorithm allows the model to adjust to time-varying transformations provided that sufficient "ground-truth" data can be provided; the precipitation case study indicates that the amount of such data required might not be overly burdensome, depending on the application. In the case of precipitation estimation, we are currently testing the methodology at a number of different sites, including the Florida and Amazon basin regions. By insufficiency of information, we mean that the available remotely sensed variables are not, in themselves, adequate for establishing a unique input-output mapping. By its nature, the MCPN methodology allows us to use as many sources of information as are considered to be pertinent to the estimation problem. For example, we have used both IR and VIS imagery in the precipitation estimation example presented here. In ongoing work we are exploring the usefulness of other remotely sensed frequency channels. Insufficiency of data is, of course, a continual problem in science. In the precipitation estimation study reported here, we used data collected over Japan by the GPCP AIP-1 specifically for the purpose of algorithm development. Although the temporal extent of the data is limited to only 2 months, the data are likely to be relatively good quality compared to those available for much of the world; rain-gauge densities tend to be sparse and the data difficult to obtain, while radar data tend to be limited to first-world countries. Therefore we are exploring methods to take advantage of alternative sources of "truth" data such as the twice-daily precipitation estimate snapshots provided by specialized microwave sensors on board polar-orbiting satellites.

From the model identification point of view, another issue of interest concerns the proper choice of image features to be used as inputs to the MCPN model. Obviously, this requires some combination of scientific insight and artistry on the part of the modeler. In the precipitation example we selected the 10 features on the basis of some preliminary explorations and the results of an earlier published study [*Hsu et al.*, 1997] in which only IR imagery was used. In ongoing work we are exploring the use of a wide variety of different input features from multiple sources of information such as, for example, topography, wind direction, relative humidity, and other relevant variables, including both observation data and estimates provided by numerical weather prediction models. A clear advantage of the MCPN model approach is its ability to easily incorporate a wide variety of such sources of relevant information in the service of improved estimates of the dependent variable. Of course, an ultimate goal would be to identify a general model which, once trained, will provide stable estimates of the dependent variable and not require constant updating.

A third, and rather critical, issue, from the hydrologic studies point of view, is that of the spatial and temporal resolution and coverage of precipitation estimates obtained from satellite-based remotely sensed data. Ground-based radar estimates of precipitation are now produced routinely for much of the continental United States at 4×4 km spatial resolution and 15–30 min temporal resolution but are only sparsely available for much of the globe. In contrast, infrared imagery collected by geostationary satellites is 8×8 km at 30–60 min temporal resolution and is available for most of the globe. Therefore, satellite-based precipitation estimates are likely to be primarily useful for hydrologic studies at large, regional, and basin scales; that is, general circulation models, numerical weather prediction models, and water balance studies. However, even this coarse resolution may be better than what is available for many parts of the globe where gauge densities are low. Further, synthesis of precipitation estimates from multiple sources could perhaps contribute to improved coverage and accuracy. Clearly, there is much scope and potential for further improvements in this area.

Finally, we believe that the MCPN model structure is such that it should readily lend itself to the estimation of various physical variables from remotely sensed data, such as surface soil moisture, surface skin temperature, vegetation parameters, etc. We welcome discussion and potential collaborative research leading to the development and testing of such applications.

Appendix: Illustrative Training Example

The process of constructing an MCPN model is demonstrated by using it to approximate the simple "Mexican Hat" function $z = f(x_1, x_2)$ shown in Figure 6a; the input variables in this case are the coordinates x_1 and x_2 defined on $x_1 \in [0, 1]$ and $x_2 \in [0, 1]$. The SOFM for the input data as constructed using 64 hidden nodes arranged in an 8×8 matrix, and the LLOM was constructed using a neighborhood Ω size of 3×3 hidden nodes. The MCPN model was trained using a test data set constructed by generating 1000 data points uniformly in the input space. The evolution of the SOFM weights is illustrated in Figures 6d–6f, where the two axes of each plot correspond to the w_{j1} and w_{j2} weight/input values defining the cluster centers of each of the $j \in 8 \times 8$ ($= 64$) nodes. Notice that the weight values begin with random initial values near the (0.5, 0.5) region of the input space and gradually spread out to uniformly span the entire input space. The function approximation performance of the original CPN and the new MCPN methods is shown in Figures 6b and 6c. The original CPN method results in a piecewise-constant approximation (Figure 6b) with a root mean square error (RMSE) = 0.094, while the MCPN method produces a superior piecewise-linear approximation (Figure 6c) with a much smaller RMSE = 0.035.

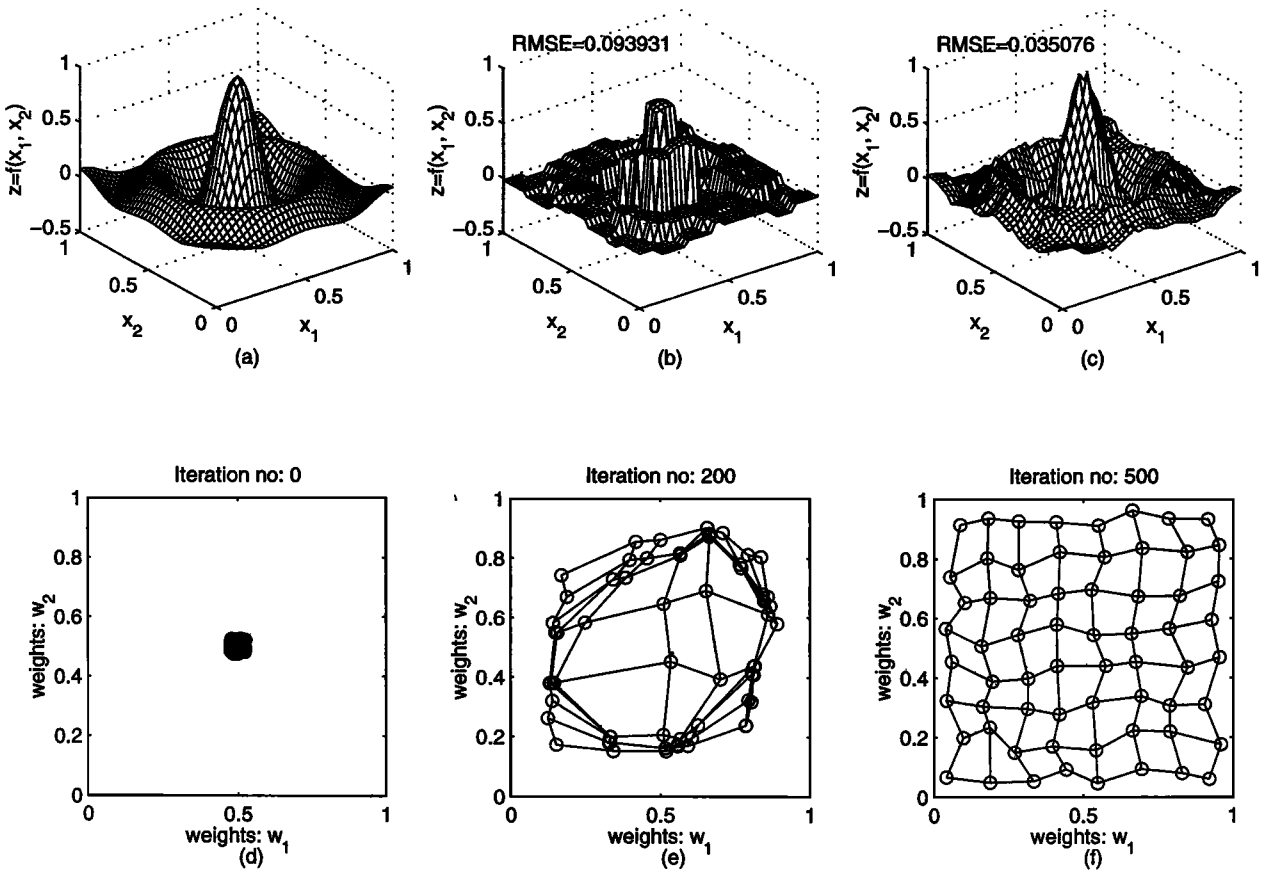


Figure 6. Function approximation example: (a) the Mexican Hat function, (b) approximation using counterpropagation network (CPN), (c) approximation using modified CPN (MCPN), (d) initial distribution of self-organizing feature map (SOFM) weights, (e) weight distribution after 200 iterations, and (f) weight distribution after 500 iterations.

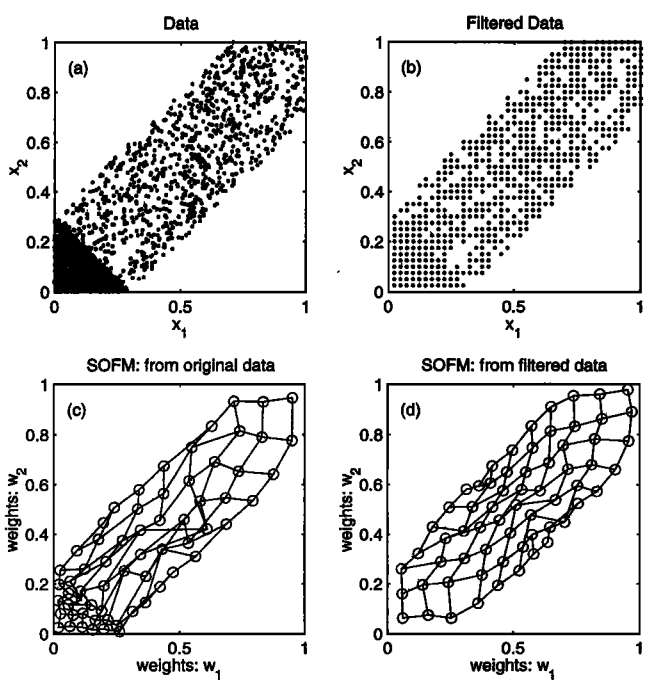


Figure 7. Example of data filtering: (a) distribution of the original data in the (x_1, x_2) input space, (b) filtered data using representative grid center point, (c) SOFM weight distribution using original data, and (d) SOFM weight distribution using filtered data.

To demonstrate the significance of data preprocessing, we reran the same example with two additional complications (see Figure 7a). First, the input variables are restricted to belong to the following diagonal strip-shaped subregion of the input space:

$$\begin{aligned} x_1 - x_2 &\leq 0.3 \\ x_1 - x_2 &\geq -0.3 \\ x_1 &\geq 0.0 \\ x_2 &\geq 0.0 \end{aligned} \tag{A1}$$

Second, the available input data set is assumed to have a higher density in the small triangular region close to the origin. The filtered data set is shown in Figure 7b after having been preprocessed to a 0.025 × 0.025 resolution grid. The SOFMs obtained for the unfiltered and filtered data sets are shown in Figures 7c and 7d, respectively. Note that the SOFM nodes for the unfiltered case are distributed in a manner reflecting the higher density of data near the origin (Figure 7c). As a result, the precision of the input-output mapping is superior in this region compared to the rest of the space. In contrast, the SOFM nodes for the filtered case are distributed uniformly over the input space (Figure 7c), resulting in a uniform mapping accuracy over the data space (the RMSE was improved from 0.043 for the unfiltered case to 0.038 for the filtered case). In neither of these cases has the procedure placed nodes in the region of the input space that is devoid of data.

Acknowledgments. This research was supported by the NOAA-PACS Research Program and the NASA-EOS Interdisciplinary Research Program. The authors appreciate Dan Braithwaite for providing the programming required to access the data and Corrie Thies for her care in proofing and editing the manuscript.

References

Adler, R. F., and A. J. Negri, A satellite infrared technique to estimate tropical convective and stratiform rainfall, *J. Clim. Appl. Meteorol.*, *27*, 30–51, 1988.

Arkin, P. A., The relationship between fractional coverage of high cloud and rainfall accumulations during GATE over the B-scale array, *Mon. Weather Rev.*, *107*, 1382–1387, 1979.

Arkin, P. A., and P. Xie, The global precipitation climatology project: First algorithm intercomparison project, *Bull. Am. Meteorol. Soc.*, *75*, 401–419, 1994.

Bischoff, H., W. Schneider, and A. J. Pinz, Multispectral classification of Landsat images using neural networks, *IEEE Trans. Geosci. Remote Sens.*, *30*(3), 482–490, 1992.

Cabrera-Mercader, C. R., and D. H. Staelin, Passive microwave relative humidity retrievals using feedforward neural networks, *IEEE Trans. Geosci. Remote Sens.*, *33*(6), 1324–1328, 1995.

Davis, D. T., Z. Chen, L. Tsang, J. N. Hwang, and A. T. C. Chen, Retrieval of snow parameters by iterative inversion of a neural network, *IEEE Trans. Geosci. Remote Sens.*, *31*(4), 842–852, 1993.

Funahashi, K. I., On the approximation realization of continuous mappings by neural networks, *Neural Networks*, *2*, 183–192, 1989.

Gori, M., and A. Tesi, On the problem of local minimum in backpropagation, *IEEE Trans. Pattern Anal. Mach. Intel.*, *14*, 76–86, 1992.

Grassotti, C., and L. Grand, Classification-based rainfall estimation using satellite data and numerical forecast model fields, *J. Appl. Meteorol.*, *33*, 159–178, 1994.

Griffith, G. C., W. L. Woodley, P. G. Grube, D. W. Martin, J. Stout, and D. N. Sikdar, Rain estimation from geosynchronous satellite imagery-visible and infrared studies, *Mon. Weather Rev.*, *106*, 1153–1171, 1978.

Grossberg, S., Emvedding field: A theory of learning with physiological implications, *J. Math. Psychol.*, *6*, 209–239, 1969.

Gupta, H. V., K. Hsu, and S. Sorooshian, Superior training of artificial neural networks using weight-space partitioning, paper presented at 1997 IEEE International Conference on Neural Networks, Inst. of Electr. and Electron. Eng., Houston, Tex., 1997.

Hecht-Nielsen, R., Counterpropagation networks, in *Proceedings of the International Conference on Neural Networks*, vol. II, pp. 19–32, IEEE Press, Piscataway, N. J., 1987.

Hecht-Nielsen, R., Applications of counterpropagation networks, *Neural Networks*, *1*, 131–139, 1988.

Hecht-Nielsen, R., *Neurocomputing*, Addison-Wesley, Reading, Mass., 1990.

Heermann, P. D., and N. Khazenie, Classification of multispectral remote sensing data using a back-propagation neural network, *IEEE Trans. Geosci. Remote Sens.*, *30*(1), 81–88, 1992.

Hornik, K., M. Stinchcombe, and H. White, Universal approximation of an unknown mapping and its derivatives using multi-layer feedforward networks, *Neural Networks*, *3*, 551–560, 1990.

Hsu, K., H. V. Gupta, and S. Sorooshian, Artificial neural network modeling of the rainfall-runoff process, *Water Resour. Res.*, *31*, 2517–2530, 1995.

Hsu, K., X. Gao, S. Sorooshian, and H. V. Gupta, Precipitation estimation from remotely sensed information using artificial neural networks, *J. Appl. Meteorol.*, *36*, 1176–1190, 1997.

Jacobs, R. A., Increased rates of convergence through learning rate adaptation, *Neural Networks*, *1*, 295–307, 1988.

King, P. W. S., W. D. Hogg, and P. A. Arkin, The role of visible data in improving satellite rain-rate estimates, *J. Appl. Meteorol.*, 34, 1608–1621, 1995.

Kohonen, T., Self-organized formation of topologically correct feature maps, *Biol. Cybern.*, *43*, 59–69, 1982.

Kosko, B., *Neural Networks for Signal Processing*, Prentice-Hall, Englewood Cliffs, N. J., 1992.

Li, L., J. Vivekanandan, C. H. Chen, and L. Tsang, Microwave radiometric technique to retrieve vapor, liquid and ice, I, Development of a neural network-based inversion method, *IEEE Trans. Geosci. Remote Sens.*, *35*(2), 224–236, 1997.

Negri, A. J., and R. F. Adler, Infrared and visible satellite rain estimation, I, A grid cell approach, *J. Clim. Appl. Meteorol.*, *26*, 1553–1564, 1987a.

Negri, A. J., and R. F. Adler, Infrared visible satellite rain estimation, II, A cloud definition approach, *J. Clim. Appl. Meteorol.*, *26*, 1565–1576, 1987b.

Negri, A. J., and R. F. Adler, An intercomparison of three satellite infrared rainfall techniques over Japan and surrounding waters, *J. Appl. Meteorol.*, *32*, 357–373, 1993.

Negri, A. J., R. F. Adler, and P. J. Wetzel, Rainfall estimation from satellites: An examination of Griffith-Woodley technique, *J. Clim. Appl. Meteorol.*, *23*, 102–116, 1984.

Serpico, S. B., and F. Roli, Classification of multisensor remote-sensing images by structured neural networks, *IEEE Trans. Geosci. Remote Sens.*, *33*(3), 562–578, 1995.

Suykens, J. A. K., J. Vandewalle, and B. De Moor, *Artificial Neural Networks for Modeling and Control of Non-linear Systems*, Kluwer Acad., Norwell, Mass., 1996.

Tsang, L., Z. Chen, S. Oh, R. J. Marks II, and A. T. C. Chang, Inversion of snow parameters from passive microwave remote sensing measurements by neural network trained with a multiple scattering model, *IEEE Trans. Geosci. Remote Sens.*, *30*(5), 1015–1023, 1992.

Tsonis, A. A., and G. A. Isaac, On a new approach for instantaneous rain area delineation in the midlatitudes using GOES data, *J. Clim. Appl. Meteorol.*, *24*, 1208–1218, 1985.

Vemuri, V. R., and R. D. Rogers (Eds.), *Artificial Neural Networks: Forecasting Time Series*, IEEE Comput. Soc. Press, Los Alamitos, Calif., 1994.

Xiao, R., and V. Chandrasekar, Development of a neural network based algorithm for rainfall estimation from radar observations, *IEEE Trans. Geosci. Remote Sens.*, *35*(1), 160–171, 1997.

Yhann, S. R., and J. J. Simpson, Application of neural networks to AVHRR cloud segmentation, *IEEE Trans. Geosci. Remote Sens.*, *33*(3), 590–604, 1995.

Yoshida, T., and S. Omatu, Neural network approach to land cover mapping, *IEEE Trans. Geosci. Remote Sens.*, *32*(5), 1103–1109, 1994.

X. Gao, H. V. Gupta, K.-L. Hsu, and S. Sorooshian, Department of Hydrology and Water Resources, University of Arizona, Building 11, P. O. Box 210011, Tucson, AZ 85721. (gao@hwr.arizona.edu; hoshin@hwr.arizona.edu; hsu@hwr.arizona.edu; soroosh@hwr.arizona.edu)

(Received July 20, 1998; revised January 25, 1999;
accepted January 26, 1999.)